# 1

# Introduction to Microsoft Smartphone

COPYRIGHTED MATERIAL

Mobile computing is everywhere today. The number of cell phones, PDAs, and other handheld mobile devices has exceeded the number of computers in the world. The Yankee Group estimates that there are approximately 1.8 billion mobile devices in use worldwide, used for a variety of tasks, including the still predominant voice communication, text messaging, web surfing, e-mail, gaming, and so on. As wireless technologies such as 3G cellular systems, wireless LANs, Bluetooth, WiMAX, and Ultra-Wideband continue to mature, empowering those mobile devices and the network infrastructure, we soon will be able to enjoy the power of mobile computing anywhere, anytime, and on any devices.

What will ultimately cause this vision to materialize are new mobile applications and services built on the pervasive computing infrastructure, along with a set of software platforms and programming tools. Motivated by this vision, we decided to write a book on one of the most powerful software platforms for mobile computing: the Microsoft Smartphone. This chapter begins with a brief introduction to smartphones and the challenges of developing smartphone applications. Then we introduce Microsoft Windows Mobile and some related technologies. Readers will likely find this section informative, as it clarifies some terms and concepts pertinent to Microsoft Smartphone.

## What Is a Smartphone?

The worldwide mobile wireless industry is quickly moving from traditional, voice-based cellular phone services to combined voice and data services, as a result of increasing demand for mobile data access and the deployment of high-speed wireless data services utilizing a variety of wireless technologies. For example, 2.5G/3G wireless services are being rolled out and used by a rapidly growing number of subscribers, and the number of WiFi hotspots and residential wireless LANs continues to grow substantially. The trend is clear: Cell phones, PDAs, and portable consumer

electronic devices will likely merge into a single, handheld device as a universal personal communicator and computing platform (generally called a *smartphone*). Indeed, the market has seen a dramatic increase in smartphone sales when compared to the fairly slow growth of PDA sales worldwide.

Generally, a smartphone is a powerful, multi-function cell phone that incorporates a number of PDA functionality, such as a personal scheduler, calendar, and address book, as well as the ability to access Internet services and applications using either a keypad or a stylus. In addition to making a call from a smartphone, users can surf the web, check e-mail, create documents, play online games, update schedules, or access an enterprise network via a virtual private network (VPN). Wireless Internet access is enabled by means of cellular wireless networks — such as GSM/GPRS, CDMA, CDMA2000, or WCDMA, among others.

Bill Gates, Chairman and then Chief Software Architect of Microsoft, introduced his vision for the smartphone at the 2004 Mobile Developer Conference:

> *The pocket devices, phone and PDA, really the trend is to have the best of both together. The phone is no longer just a voice-only device; more and more it has that rich, color screen. A PDA is no longer a disconnected device; more and more it's got the ability to make calls and connect up to wireless data networks. In many cases that will be both the wide area data networks, 2 ½ G or 3G networks, but also increasingly you'll have WiFi connectivity built into the device as well. So it will be able to connect up to whichever network is available, whichever one provides the best bandwidth and economics there.*

An increasing number of high-end cell phones and smartphones are equipped with powerful mobile processors (such as ARM processors), 64–128MB memory, 256–512MB flash storage, and even 2–4GB hard drives. Examples include the Motorola Q, SPV C600, and O2 XDA II. Some smartphones are PDA-based with handwritten recognition or a tiny keyboard, and phone functions as add-on features, such as Palm Xplore and Palm Treo. In fact, cell phone manufacturers and PDA manufacturers have different views regarding the future of smartphone devices. Unsurprisingly, each camp believes their device will prevail, with add-on functionality of devices from the other side. As wireless technologies and the mobile market continue to evolve, it is still too early to tell which approach will finally win. Nevertheless, one thing is certain: They both need reliable, high-performance, low-power consumption operating systems and software to leverage the wireless services.

Microsoft Smartphone refers to Microsoft's platform for next-generation cell phones — basically a software architecture with Windows CE as the operating system, plus a rich set of applications such as Pocket Internet Explorer and Pocket Outlook and powerful software development tools such as .NET Compact Framework and Visual Studio 2005. (We use *Smartphone* to refer to Microsoft Smartphone throughout this book and *smartphone* to refer to general multifunction cell phones.)

This book focuses on software development issues and practices on smartphones running Microsoft Windows Mobile software. There are, of course, other software development solutions. For example, Palm Inc., also provides a software development kit (SDK) for Palm OS smartphones, and you can find an SDK and supporting tools for Symbian OS, another popular cell phone operating system.

# Smartphone Applications and Services

With the vision of mobile convergence supporting communication and computing on a single set of hardware components, mobile wireless network operators, cell phone manufactures, and independent

software vendors are working together to create new applications and services with the hope of taking a lead position in the next wave of mobile computing. These services and applications essentially leverage the increasingly high computing capability supplied by the cell phone and the flexible, high-speed wireless connectivity to offer an efficient, reliable, and rich experience to the end user. This section summarizes the potential services and applications in this domain.

## Mobile Commerce

This category includes mobile banking, location-based business information service and shopping assistance, mobile advertising, and mobile payment, among other services. Japan and Korea already offer widespread mobile payment applications that enable consumers to make purchases at a convenience store by waving the cell phone past a reading device. Numerous startup companies in the United States are developing applications that enable credit card payments to be verified, parking fees to be paid at the meter, and social networking. Industries involved in this category include banks, credit card companies, retail stores, stock trading agencies, and online businesses.

## Mobile Enterprise

Services and applications in this category are concerned with mobile worker assistance such as real-time job scheduling, route planning, package delivery updates, mobile collaboration and communication, and mobile business transaction. Moreover, enterprise resource planning (ERP) applications and supply chain management (SCM) systems can be extended to support mobile access and onsite processing. In addition to mobile enterprise, law enforcement, educational, and healthcare organizations may also utilize these services and solutions to improve productivity and reduce costs.

### Other Mobile Software Platforms

Symbian OS is developed by Symbian, a company supported by several cell phone manufacturers, including Nokia, Ericsson, Sony Ericsson, and Samsung. Originally based on the EPOC operating system, Symbian OS defines several UI reference models for different types of devices. Symbian OS uses EPOC C++, a pure object-oriented language, as the supporting programming language for both system services implementations and APIs. It also allows Java applications for mobile devices (Java 2 Micro Edition, J2ME, applications) to run on top of a small Java runtime environment. The Symbian Developer website (`www.symbian.com/developer`) provides numerous technical documents for Symbian OS, SDKs, and sample code, as well as information on Symbian OS development and the Symbian developer community.

Palm OS, developed by Palm Inc., is a preemptive, multitasking operating system for Palm PDAs and cell phones. Palm OS supports both the ARM and Motorola 68000 architectures. Developers can choose a programming language from C, C++, Visual Basic, or Java, although C is most widely used for Palm OS software development. Interested readers can visit the Palm OS developer site (`www.palmsource.com/developers`) for more technical details. Palm OS application development is facilitated by the Palm OS 68K and Protein SDKs and some commercial developer suites. A developer suite is an integrated software tool that enables developers to create both ARM-native and Palm OS Protein-powered applications for Palm OS Cobalt and 68K applications.

### *Mobile Data Service and Entertainment*

This category includes real-time, location-based navigation assistance coupled with traffic data access, mobile gaming, rich media services, and so on. Mapping and GPS-based navigation services are increasingly being integrated into general-purpose smartphone platforms. Mobile television services have been available in the United States and some Asian countries; online music download services (such as Apple's iTunes service) are available on some high-end smartphones; mass media companies, music and movie companies, online gaming service providers, and of course the consumer, will be involved in this category of services and applications.

Needless to say, the aforementioned summary is by no means exhaustive; however, it is indicative of the broad range of new services and applications with tremendous potential for businesses. Indeed, the enormous opportunity of next-generation mobile computing has created myriad services and applications that will likely continue to mature and succeed in the foreseeable future. What all these services and applications share is a reliance on software running on smartphones to reach the end user. To this end, software designers and developers have to be aware of the challenges and obstacles involved in smartphone-based application development.

## Challenges of Smartphone Application Development

The application design paradigm for smartphones not only differs largely from that of desktop applications, it also has some inherent requirements that separate it from application development on common mobile devices. First, the hardware constraints of a smartphone, such as processor speed, persistent storage capacity, battery, and wireless connection, significantly affect the application design principles. Second, the input method of a smartphone, either a telephone keypad with additional navigation keys or a soft keyboard, forces application developers to pay more attention to the GUI of a smartphone application than that of desktop applications. Third, the cost of wireless data service remains a major factor for mass adoption of Smartphone technology, as it is still quite high compared with the cost of landline Internet services. Moreover, the application must be easily ported to various hardware platforms using different processor architectures and peripherals. Following are the salient factors to consider while developing applications for smartphones:

- **Efficient storage and adaptive networking.** Despite the advancements in mobile embedded and wireless technologies, smartphones usually have neither a large memory capacity nor a reliable, high bandwidth wireless connection. It is crucial, therefore, to make better use of precious local storage for high performance. The networking functionality of a smartphone must be able to adapt to the comparatively low bandwidth and high drop rate of wireless links. Finally, remote data access must be able to function in disconnected mode. For a more detailed discussion of data access on a smartphone, see Chapters 5 and 6.
- **Simple and user-friendly GUI.** Because entering characters on a smartphone is not as easy as typing on a PC keyboard (although some teenagers will disagree), a program's GUI has to deal with most of the input by using graphical components. Due to a smartphone's relatively small screen, the layout of these components on a window, as well as the grouping of components into different windows, has to been designed carefully. For example, you generally should eliminate horizontal scrolling for a smartphone application. GUI design is discussed in more detail in Chapter 4.

- **Lightweight computation and power management.** A smartphone is not designed for CPU-intensive applications; instead, most CPU-intensive computation should be done on the server side, which is typically much more powerful than a mobile processor, of an application whenever possible, enabling the client side on a smartphone to run faster. This creates a scenario for peer-to-peer mobile applications, whereby mobile nodes essentially serve each other without relying on a central server. Web services is a good example of offloading computation to the server. Chapter 9, "XML and Web Services," provides more details about accessing web services on a smartphone. Power management is also crucial to increasing battery life. Smartphone operating systems must be able to put the device into power-save state whenever possible (subject to a user's configuration) to save battery power.
- **High security.** In a sense, a smartphone can be viewed as another form of personal identification, with sensitive data stored and transmitted over the air. There have been a number of cell phone–related security hacks utilizing various vulnerabilities of cell applications. As the smartphone evolves to function not only as a telephone but also as a credit card or ID card, the compromise of a cell phone will result in serious problems. Low-overhead authentication and authorization must be enforced for critical smartphone applications. Chapter 12, "Device and Application Security," and Chapter 13, "Data and Communication Security," cover smartphone security issues.

# Introducing Microsoft Windows Mobile

To facilitate the design and development of smartphone applications, hardware manufactures and software vendors have teamed up to provide powerful programming tools for smartphone developers. One of the solutions is Microsoft's Windows Mobile, a unified platform specifically designed to enable developers to leverage Windows desktop programming experiences for mobile application design and development targeting both smartphones and PDAs. Because of the innovative .NET Compact Framework and bundled programming tools, Microsoft Windows Mobile for Smartphone has gained a sizable amount of popularity among software developers worldwide.

Windows Mobile is Microsoft's software platform for Pocket PCs (PDAs running Microsoft's software platforms) and Smartphones (smartphones running Microsoft's software platform). Here the notion of "software platform" refers to an integrated computing environment that consists of an operating system, the .NET runtime environment, a set of applications, and related application development tools. The concepts of "Pocket PC" and "Smartphone" are also commonly referred to as software platforms targeting PDAs and smartphones, respectively.

## *Windows Mobile 5.0*

The latest version of Microsoft Windows Mobile is Windows Mobile 5.0. Windows Mobile consists of a tailored Windows CE operating system; the Microsoft .NET Compact Framework (the runtime and the class libraries); a set of tools and APIs for native code development; a device software emulator; and for developers, an IDE (Integrated Development Environment) component for Visual Studio 2005.

Windows Mobile 5.0 is based on the Windows CE .NET 5.0 operating system. As part of the Smartphone offering, Microsoft also provides the Smartphone 2005 SDK for ISVs (independent software vendors). Developers can use the Smartphone SDK in conjunction with Visual Studio 2005, the flagship programming environment tool from Microsoft, for .NET Compact Framework–based managed code application

development. The Smartphone SDK also allows C or C++ unmanaged application programming utilizing Win32 APIs using Microsoft embedded Visual C++ (eVC). The next version of Windows Mobile is called Windows Mobile 6.0 and code named Photon.

Windows Mobile 5.0 unifies application development for Smartphones and Pocket PCs by providing the same Win32 APIs and a set of development tools. Unlike the previous version of Windows Mobile, Windows Mobile 2003, which does not allow developers to use SQL Server CE in Smartphone applications, now both Pocket PC and Smartphone platforms can take advantage of the same set of common data services provided by SQL Server 2005 Mobile Edition, the next release of SQL Server CE. In addition, both platforms share the same security model and the same common application installer. The unification of these two platforms reflects Microsoft's mobile platform strategy in response to the imminent convergence of these two types of mobile devices.

In addition, Windows Mobile 5.0 provides a new set of standardized WinCE APIs, including multimedia APIs supporting Windows Media 10 Mobile, Direct 3D Mobile APIs, new GPS APIs, DirectDraw APIs, and camera APIs. These APIs essentially extend the functionality of a Windows Mobile device and offer an improved user experience. On the managed application environment side, Windows Mobile 5.0 comes with .NET Compact Framework 2.0 and provides a set of .NET managed APIs for messaging and telephony, which are not available in .NET 1.0 and 1.1. Chapter 2 discusses the .NET Compact Framework in detail.

Application development on Windows Mobile 5.0 can further take advantage of the latest release of Microsoft's flagship IDE, Visual Studio 2005. The software emulator in Visual Studio 2005 has been rewritten to eliminate the gap between emulation and physical device deployment. In addition, both managed and unmanaged code can be developed within the same Visual Studio 2005 environment. No eVC is needed anymore. Additionally, Visual Studio 2005 includes a number of improvements in the form designer and deployment tools, as well as common programming and debugging support. This book uses Visual Studio 2005 as the application development platform, so you will see screenshots taken from it and code developed using a set of smart device tools integrated into Visual Studio 2005.

# Microsoft Smartphone from a User's Perspective

The market for cell phone operating systems is highly segmented. Many cell phones run operating systems other than Microsoft Smartphone; indeed, Microsoft is relatively new to the cell phone platform market. The stronghold in this market is Symbian OS, which is funded and supported by cell phone makers such as Nokia, Sony Ericsson, Siemens, and others. Palm is historically strong in the PDA market. For people with a background in those types of platforms, it is necessary to present a quick tour into Microsoft Smartphone. Those of you who have used a Smartphone device for some time can feel free to skip this section.

To a typical user, a Smartphone is a much more powerful cell phone. It provides many more types of applications, and the user interface is more sophisticated than a traditional cell phone. Like most other cell phones, a Microsoft Smartphone actually refers to a combination of the handset and its running applications. The aesthetic design of a smartphone handset may vary significantly, but the principle user interface — how a user interacts with the device — is almost the same across different devices from different manufacturers. Let's take a look at the Smartphone emulator, a software tool that helps developers quickly develop and test Smartphone applications without using a physical device. Details about the Smartphone emulator can be found in Chapter 3.

Figure 1-1 depicts a common layout of a Smartphone device. Although color doesn't appear in these screenshots, a color screen at the top and the number keys at the bottom are the most common elements. What separates a Smartphone from a PDA are the software keys. Notice the two soft keys (left and right) directly below the screen. These correspond to the menu bar and commands at the bottom of the screen. Depending on how applications define the functions of the menu bar, these soft keys may perform specific tasks. Also note the five-way (up, down, left, right, and "Select") navigation pad and the four fixed function keys: Call, Home, Hang Up, and Back. Pressing the Call key will bring you to the phone call screen, where you can enter a phone number. After you enter a phone number, pressing the Call key again will initiate the call. The Hang Up key, of course, is used when you want to end a call. The Home key always takes you to the home screen shown in this figure, and the Back key enables you to go back to the previous screen. At the home screen, pressing any number key will automatically bring you to the phone call screen.

**Figure 1-1**

A Smartphone usually has a power button, a record button (for voice recording), and two volume control buttons. Some Smartphones have a built-in camera, so there will be another button for it (some have a high-quality digital camera, such as Nokia N90, which features a Carl Zeiss lens, 2-megapixel resolution, and a 20x zoom).

The first-generation Smartphone applications are mostly clones of desktop Windows applications, enabling users familiar with those desktop applications to avoid learning a new one. Typical Smartphone applications are Pocket Outlook, calendar, contacts, Pocket Word, Pocket Excel, Windows Media Player, MSN messenger, games, and some accessories. The second-generation Smartphone applications are exclusively designed to leverage the advantages of mobility and ubiquitous wireless access. In the next several years we will see a whole new set of applications that utilize real-time location information in conjunction with always-on wireless data access.

Overall, the design of a Smartphone aims to take advantage of a user's prior experience with Windows desktop systems. After all, a Smartphone is a small computer running a stripped-down version of the Windows operating system. This greatly helps users familiar with Palm or Symbian cell phones because the learning curve is largely eliminated.

# Summary

This chapter introduced the concept of the smartphone and the trend of convergence in the mobile computing and communication domain. As a converged mobile device, smartphone, in conjunction with supporting new mobile wireless services and applications, will gain widespread popularity worldwide in the foreseeable future. Microsoft Windows Mobile is a software platform that enables fast, efficient, and feature-rich application development for Smartphone devices. Many smartphone and cell phone manufacturers (including Palm, which traditionally uses Palm OS on their PDAs and smartphones) have started to use Windows Mobile as the underlying software platform on their products, largely because users can leverage their knowledge of Windows desktop systems in using a Smartphone.

Beginning with the next chapter, our discussion moves to the technical foundations of Windows Mobile. We will focus on the core components of the .NET Compact Framework, as well as the class libraries and type systems of .NET. In addition, a primer on the C# programming language will be provided.